|  |  |
| --- | --- |
| 7.1 | Twenty five numbers are entered from the keyboard into an array. Write a program to find out how many of them are positive, negative, how many are even and odd. |
| Code | Text  Description automatically generated |
| Output | Text  Description automatically generated |
|  |  |
| 7.2 | Write a program for creating two arrays of different size and merge both arrays into one by sorting  those arrays in ascending order. [Merge by sorting] |
| Code | #include<stdio.h>  int main()  {  int i,j,k,m,n,temp;  int a[5]={1,2,5,3,4};  int b[2]={7,6};  int c[7];  for(i=0;i<5;i++)  {  c[i]=a[i];  }  k=i;  for(i=0;i<2;i++)  {  c[k]=b[i];  k++;  }  for(i=0;i<k;i++)  {  printf("%d\n",c[i]);  }  for(m=0;m<7;m++)  {  for(n=m+1;n<7;n++)  {  if(c[m]>c[n])  {  temp=c[m];  c[m]=c[n];  c[n]=temp;  }  }  }  printf("After sorting : \n");  for(m=0;m<7;m++)  {  printf("%d\n",c[m]);  }  return 0;  } |
| Output | Text  Description automatically generated |
|  |  |
| 7.3 | Write a program to multiply any two 3\*3 matices. |
| Code | #include<stdio.h>  int main(){  int mat31[10][10], mat32[10][10], mult[10][10];  int i, j, k;  printf("\n Enter the First Matrix elements : \n"); // Read 1st Matrix  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  scanf("%d", &mat31[i][j]);  }  }  printf("\n Enter the Second Matrix elements : \n"); // Read 2nd Matrix  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  scanf("%d", &mat32[i][j]);  }  }  printf("\n The First Matrix : \n\n"); // Print 1st Matrix  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  printf("\t %d ", mat31[i][j]);  }  printf("\n");  }  printf("\n The Second Matrix : \n\n"); // Print 2nd Matrix  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  printf("\t %d ", mat32[i][j]);  }  printf("\n");  }  // . . . The two matrices have been multipled here . . .  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  mult[i][j] = 0;  for (k = 1; k <= 3; k++){  mult[i][j] = mult[i][j]+ mat31[i][k] \* mat32[k][j];  }  }  }  // . . . Product of the given two Matrices is displayed in Matrix form . . .  printf("\n\n The Resultant of Matrix Multiplication is \n\n");  for (i = 1; i <= 3; i++){  for (j = 1; j <= 3; j++){  printf("\t%d", mult[i][j]);  }  printf("\n");  }  return (0);  } |
| Output | Text  Description automatically generated |
|  |  |
| 8.1 | Take a user input for a string and calculate the number of alphabets, digits and special  Characters from the given input. |
| Code | #include <stdio.h>  #define MAX\_SIZE 100 // Maximum string size  int main()  {  char str[MAX\_SIZE];  int alphabets, digits, others, i;  alphabets = digits = others = i = 0;  /\* Input string from user \*/  printf("Enter any string : ");  gets(str);  /\*  \* Check each character of string for alphabet, digit or special character  \*/  while(str[i]!='\0')  {  if((str[i]>='a' && str[i]<='z') || (str[i]>='A' && str[i]<='Z'))  {  alphabets++;  }  else if(str[i]>='0' && str[i]<='9')  {  digits++;  }  else  {  others++;  }  i++;  }  printf("Alphabets = %d\n", alphabets);  printf("Digits = %d\n", digits);  printf("Special characters = %d", others);  return 0;  } |
| Output | Text  Description automatically generated |
|  |  |
| 8.2 | Write a program that takes a set of names of individuals and abbreviates the first middle  and other names except the last name by their first letter. |
| Code | #include<stdio.h>  void main()  {  char name[9][20];  printf("Enter the first name:");  gets(name[0]);  printf("Enter the middle name:");  gets(name[1]);  printf("Enter the last name:");  gets(name[2]);  printf("Enter the first name:");  gets(name[3]);  printf("Enter the middle name:");  gets(name[4]);  printf("Enter tha last name:");  gets(name[5]);  printf("Enter the first name:");  gets(name[6]);  printf("Enter the middle name:");  gets(name[7]);  printf("Enter the last name:");  gets(name[8]);  printf("The abbreviate names are as follows:\n");  printf("1. %c. %c. %s\n",name[0][0],name[1][0],name[2]);  printf("2. %c. %c. %s\n",name[3][0],name[4][0],name[5]);  printf("3. %c. %c. %s\n",name[6][0],name[7][0],name[8]);  } |
| Output | Text  Description automatically generated |
|  |  |
| 8.3 | Write a program to check if the user inputted string is palindrome or not  using recursion |
| Code | #include<stdio.h>  #include<string.h>  int main()  {  char str[100];  int i, len, flag;  flag = 0;    printf("\n Please enter any string : ");  gets(str);    len = strlen(str);    for(i = 0; i < len; i++)  {  if(str[i] != str[len - i - 1])  {  flag = 1;  break;  }  }  if(flag == 0)  {  printf("\n %s is a Palindrome String", str);  }  else  {  printf("\n %s is Not a Palindrome String", str);  }    return 0;  } |
| Output | Text  Description automatically generatedText  Description automatically generated |
|  |  |
| 9.1 | Write a C program to check if the entered number is prime or not by using types of  user defined functions   * No arguments passed * No arguments passed but a return value * Argument passed but no return value * Argument passes and a return value |
| Code (1) | #include<stdio.h>  #include<conio.h>  void function\_one();  void main()  {  function\_one();  }  void function\_one()  {  int number,i,flag=0;  printf("Enter the number :");  scanf("%d",&number);  for(i=2;i<=number/2;i++)  { if(number%i==0)  {  flag=1;  break;  }  }  if(flag==0)  {  printf("%d is a PRIME NUMBER.\n\n",number);  }  else  {  printf("%d is NOT A PRIME NUMBER.\n\n",number);  }  } |
| Output(1) | Text  Description automatically generatedText  Description automatically generated |
| Code(2) | #include<stdio.h>  #include<conio.h>  void function\_two(int num);  int main()  {  int number;  printf("Enter the number:");  scanf("%d",&number);  function\_two(number);  return 0;  }  void function\_two(int num)  {  int flag=0,i;  for(i=2;i<=num/2;i++)  {  if(num%i==0)  {  flag=1;  break;}  }  if(flag==0)  {  printf("%d is PRIME NUMBER.\n",num);  }  else  {  printf("%d is NOT A PRIME NUMBER.\n",num);  }  } |
| Output(2) | Text  Description automatically generatedText  Description automatically generated |
| Code(3) | #include<stdio.h>  #include<conio.h>  int function\_three();  void main()  {  int prime;  prime=function\_three();  if(prime==0)  {  printf("The number is PRIME NUMBER.\n");}  else  {  printf("The number is NOT A PRIME NUMBER.\n");  }  }  int function\_three()  {  int i,num,flag=0;  int a;  printf("Enter the number:");  scanf("%d",&num);  for(i=2;i<=num/2;i++)  {  if(num%i==0)  {  flag=1;  break;  }  }  if(flag==0)  {  a=0;  }  else  {  a=1;  }  return a;  } |
| Output(3) | Text  Description automatically generatedText  Description automatically generated |
| Code(4) | #include<stdio.h>  #include<conio.h>  int function\_four(int num);  int main()  {  int number,result;  printf("Enter the number:");  scanf("%d",&number);  result=function\_four(number);  if(result==0)  {  printf("%d is a PRIME NUMBER.\n",number);  }  else  {  printf("%d is NOT A PRIME NUMBER.\n",number);  }  return 0;  }int function\_four(int num)  {  int i,flag=0;  for(i=2;i<=num/2;i++)  {  if(num%i==0)  {  flag++;  }  }  return flag;  } |
| Output(4) | Text  Description automatically generatedText  Description automatically generated |
| 9.2 | If the length of the sides of a triangle are denoted by a, b and c then the  Area of the triangle is given by:  S = (a + b + c) / 2  A = sqrt [s(s-a)\*(s-b)\*(s-c)]  Use nested function. |
| Code | #include <stdio.h>  #include <math.h>  int main()  {  float a, b, c, s, area;  printf("Enter the length of side 1 :");  scanf("%f", &a);  printf("Enter the length of side 2 :");  scanf("%f", &b);  printf("Enter the length of side 3 :");  scanf("%f", &c);  s = (a + b + c) / 2;  area = sqrt(s\*(s - a)\*(s - b)\*(s - c));  printf("\nThe area of the triangle is %0.2f", area);  return 0;  } |
| Output | Text  Description automatically generatedText  Description automatically generated |
|  |  |
| 9.3 | A positive integer is entered through the keyboard, write a function to find the  binary equivalent of this number using recursion. |
| Code | #include<stdio.h>  int non\_rec\_bin(int num);  int rec\_bin(int num);  void main()  {  int num;  printf("Enter Number: ");  scanf("%d", &num);  printf("Decimal To Binary Using Recursion: %d", rec\_bin(num));  printf("\nDecimal To Binary Without Using Recursion: %d", non\_rec\_bin(num));  }  int non\_rec\_bin(int num)  {  int x, res=0, pos=1;  while (num!=0)  {  x = num%2;  res = res + (x\*pos);  pos = 10\*pos;  num = num/2;  }  return res;  }  int rec\_bin(int num)  {  if(num==0)  {  return 0;  }  else  {  return ((num%2)+10\*rec\_bin(num/2));  }  } |
| Output | Text  Description automatically generated |
|  |  |
| 10.1 | Write a C program to create a structure of Book Detail and display the details of the  Book in appropriate format by passing structure as argument. |
| Code | #include<stdio.h>  #include<conio.h>  #include<string.h>  struct book\_details  {  char name[50];  char book\_name[50];  char issue\_date[40];  float prize;  int roll\_no;  char phonenumber[20];  };  void details(struct book\_details student);  int main()  {  struct book\_details student;  details(student);  return 0;  }  void details(struct book\_details student)  {  printf("Enter student name: ");  gets(student.name);  printf("Enter the book name: ");  gets(student.book\_name);  printf("Enter Issued date: ");  gets(student.issue\_date);  printf("Enter the Roll No of student: ");  scanf("%d",&student.roll\_no);  printf("Enter the prize of the book: ");  scanf("%f",&student.prize);  fflush(stdin);  printf("Enter the phone number of issed student:");gets(student.phonenumber);  printf("---------------------BOOK DETAILS------------------------\n");  printf("Student name: %s\n",student.name);  printf("Book name: %s\n",student.book\_name);  printf("Issued date: %s\n",student.issue\_date);  printf("Student Roll no: %d\n",student.roll\_no);  printf("Prize : %.2f\n",student.prize);  printf("Student Phone number: %s\n",student.phonenumber);  } |
| Output | Text  Description automatically generated |
|  |  |
| 10.2 | Create a Union called library to hold accession number, title of the book, author  name, price of the book and flag indicating whether the book is issued or not  (flag = 1 if the book is issued, flag = 0 otherwise). Write a program to enter data  of one book and display the data |
| Code | #include<stdio.h>  #include<conio.h>  #include<stdlib.h>  union library  {  char book\_title[50];  char author\_name[30];  float bookPrize;  long int accession\_no;  int flag;  };  int main()  {  union library book;  printf("Enter the Book Title:");  gets(book.book\_title);  printf("BOOK TITLE : %s\n\n",book.book\_title);  printf("Enter Author name:");  gets(book.author\_name);  printf("AUTHOR NAME : %s\n\n",book.author\_name);  printf("Enter Book Prize:");  scanf("%f",&book.bookPrize);  printf("BOOK PRIZE : %.3f\n\n",book.bookPrize);  printf("Enter Accession No:");  scanf("%ld",&book.accession\_no);  printf("ACCESSION NO. : %10ld\n\n",book.accession\_no);  printf("Is the book issued (Enter 1 for YES & 0 for NO):");  scanf("%4d",&book.flag);  if(book.flag==1){  printf("BOOK ISSUED.\n");  }  else  {  printf("BOOK NOT ISSUED.\n");  }  return 0;  } |
| Output | Text  Description automatically generated |
|  |  |
| 10.3 | Write a C program for nested structure to display emplyee details such as, Age,  Name, Address, Salary. |
| Code | #include<stdio.h>  #include<conio.h>  #include<stdlib.h>  struct employee  {  char name[30];  char designation[30];  char address[60];  int age;  };struct employee1  {  struct employee details;  long int basicAllowances,houseAllowances,travelAllowances,totalSalary;  }salary;  int main()  {  printf("Enter the Employee name :");  gets(salary.details.name);  fflush(stdin);  printf("Enter the age of employee :");  scanf("%5d",&salary.details.age);  fflush(stdin);  printf("Enter the Employee Designation :");  gets(salary.details.designation);  fflush(stdin);  printf("Enter the Employee address :");  gets(salary.details.address);  printf("---Enter Salary Details---\n");  printf("Enter the Basic Allowances :");  scanf("%ld",&salary.basicAllowances);  printf("Enter the House Allowances :");  scanf("%ld",&salary.houseAllowances);  printf("Enter the Travel Allowances :");  scanf("%ld",&salary.travelAllowances);  salary.totalSalary=salary.basicAllowances+salary.houseAllowances+salary.travelAllowanc  es;  printf("-----------------EMPLOYEE DETAILS----------------\n");  printf("Employee name : %s\n",salary.details.name);  printf("Employee Age : %d\n",salary.details.age);  printf("Employee Designation :%s\n",salary.details.designation);  printf("Employee Address :%s\n",salary.details.address);  printf("Employee Basic Allowances : %ld\n",salary.basicAllowances);  printf("Employee House Allowances : %ld\n",salary.houseAllowances);  printf("Employee Travel Allowances : %ld\n",salary.travelAllowances);  printf("Employee TOTAL SALARY : %ld\n",salary.totalSalary);  return 0;  } |
| Output | Graphical user interface, text  Description automatically generated |
|  |  |
|  |  |
| 11.1 | Write a program to read the marks of 10 students for the subject CE143  Computer concepts and programming and computes the number of students  In categories FAIL, PASS, FIRST, CLASS and DISTINCTION using Pointers and  Arrays.   |  |  | | --- | --- | | Marks | Categories | | 70 or Above | DISTINCTION | | 69 to 60 | FIRST CLASS | | 59 to 40 | PASS | | Below 40 | FAIL | |
| Code | #include<stdio.h>  #include<conio.h>  #include<stdlib.h>  void main()  {  int marks[10],\*p,i,a;  int distinction=0,firstClass=0,pass=0,fail=0;  p=marks;  printf("Enter the marks of 10 Students:\n");  for(i=0;i<10;i++)  {  scanf("%d",(p+i));  }  printf("The marks of 10 students are as follows:\n");for(i=0;i<10;i++)  {  printf("Student [%d] = %d\n",i+1,\*(p+i));  }  for(i=0;i<10;i++)  {  if(\*(p+i)>=70)  {  //distinction++;  a=1;  }  else if(\*(p+i)<=69 && \*(p+i)>=60)  {  //firstClass++;  a=2;  }  else if(\*(p+i)<=59 && \*(p+i)>=40)  {  //pass++;  a=3;  }  else  {  //fail++;  a=4;  }  switch(a)  {  case 1:  distinction++;  break;  case 2:  firstClass++;  break;  case 3:  pass++;  break;  case 4:  fail++;  break;  }  }  printf("The Total DISTINCTION Students are - %4d\n",distinction);  printf("The Total FIRST CLASS Students are - %4d\n",firstClass);  printf("The Total PASS Students are - %4d\n",pass);  printf("The Total FAIL Students are - %4d\n",fail);  } |
| Output | Text  Description automatically generated |
|  |  |
| 11.2 | Write a program that counts the number of ‘e’ in the following array of pointers to strings.  (Array of Pointer)  char \*s[ ] = {  “We will teach you how to…”,  “Move a mountain”,  “Level a building”,  “erase the past”,  “Make a million”,  “…all through C!”  } ; |
| Code | #include<stdio.h>  #include<conio.h>  #include<string.h>  void main()  {  char \*str[]={"we will teach you how to...","move a mountain","level a buliding","erase  the past","make a million","...all through C!"};  char \*p;  int i,count=0;  for(i=0;i<6;i++)  {  for(p=str[i];\*p;p++)  {if(\*p=='e' || \*p=='E')  {  count++;  }  }  }  printf("we will teach you how to...\n");  printf("move a mountain\n");  printf("level a buliding\n");  printf("erase the past\n");  printf("make a million\n");  printf("...all through C!\n\n");  printf("The total \'e\' in the sentences are - %d\n",count);  } |
| Output | Text  Description automatically generated |
|  |  |
| 11.3 |  |
| Code (1) | #include<stdio.h>  void display();  int main()  {  void (\*func\_ptr)();  func\_ptr=display;  printf("Address of functions display is %u\n",func\_ptr);  (\*func\_ptr)();  return 0;  }void display()  {  puts("By helping others, we help overselves!!");  } |
| Output (1) | Text  Description automatically generated |
| Code(2) | #include<stdio.h>  char copy (char\*,char );  int main()  {  char \*str;  char source[] = "Kindness";  char target[10];  str=copy(target,source);  printf("%s\n",str);  return 0;  }  char copy(char t,char \*s)  {  char \* r;  r = t;  while(\*s!='\0')  {  \*t=\*s;  t++;  s++;  }  \*t='\0';  return(r);  } |
| Output(2) | Graphical user interface, text  Description automatically generated |